REMARKS

Claims 6-11, 17-22 and 25 are pending in the present application. No amendments to the claims are made by this Response. Reconsideration of the claims in view of the following remarks is respectfully requested.

I.   Telephone Interview

Applicants' representative contacted Examiner Yuan to schedule a telephone interview prior to filing of this Response however, it was not possible to schedule the interview prior to the 3 month response due date of the non-final Office Action. Therefore, Applicants respectfully request that the Examiner contact Applicants' undersigned representative prior to further action in the present application to schedule and conduct a telephone interview to discuss the present claims and their distinctions over the alleged combination of references.

II.   35 U.S.C. § 103, Obviousness

The Office Action rejects claims 6-11, 17-22, and 25 under 35 U.S.C. 103(a) as being allegedly unpatentable over Meltzer (U.S. Patent No. 6,226,675 B1) in view of Villacis et al., "A Web Interface to Parallel Program Source Code Archetypes," 1995, ACM, Inc., pages 1-16. This rejection is respectfully traversed.

With regard to independent claim 6, the Final Office Action states:

> Regarding independent claims 6, 17, and 25, Meltzer discloses:
> A method, data processing system, and computer program product on a computer readable medium of dynamically translating an application program into a markup language file comprising:
> parsing a document type definition file for a markup language (Meltzer on col. 23, lines 38-60: teaches parsing a document to retrieve DTD (document type));
> selecting an element defined in the document type definition file (Meltzer on col. 23, lines 38-60: teaches element retrieved from XML DTD; on col. 25, line 52 – col. 26, line 9 teaches the JAVA to XML event generator receives the stream of events from the JAVA walker and translates the selected ones to present a JAVA object as an XML

document; on col. 30, lines 55-61: teaches JAVA beans correspond to the logical structures in the DTD for transforming XML to JAVA and from JAVA to XML); and

writing the selected element to a markup language file (Meltzer on col. 23, lines 38-60: teaches producing an output by received XML element).

However, Meltzer does not explicitly disclose "executing an application program" and "a routine called by the application program".

Villacis discloses the conversion of source codes or programs into WWW hypertext documents. A special compiler is used to examine the source code and discover all subroutine call sites (routines) to automatically build the hypertext links to the appropriate subroutine definitions (see Abstract and pages 4, 7-8 and 14).

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to have modified Villacis into Meltzer to provide a program contains subroutine call sites for conversion of programs into hypertext documents, as taught by Viallacia, incorporated into the conversion system of Meltzer, in order to help programmers turn source code into hypertext documents in scalable parallel computer environment.

Office Action dated September 23, 2003, pages 2-3.

Independent claim 6, which is representative of independent claims 17 and 25 with regard to similarly recited subject matter, reads as follows:

6.   A method of dynamically translating an application program into a markup language file, the method comprising the computer-implemented steps of:

executing said application program;

parsing a document type definition file for a markup language;

during execution of said application program, selecting an element defined in the document type definition file based on a routine called by said application program; and

writing the selected element to a markup language file to form a translation.

(emphasis added)

The Meltzer reference has been discussed at length in the Response file July 3, 2003 and the Appeal Brief filed February 4, 2004, the remarks of which are hereby incorporated by reference. Therefore, Applicants will not go into a detailed discussion of Meltzer here since the many distinctions of the presently claimed invention over the

Meltzer reference have been asserted in previous filings by Applicants. Suffice it to say, Applicants respectfully submit that Meltzer does not select an element defined in a document type definition file based on a routine called by an application program during execution of the application program, as recited in claim 6. Furthermore, the Office Action agrees and admits that Meltzer does not teach executing an application program or selecting an element defined in a document type definition file based on a routine called by the application program. However, the Office Action alleges that these features are taught by Villacis. Applicants respectfully disagree.

Villacis is directed to a set of tools for annotating and exploring program source code on the world wide web. The tools in Villacis are part of a project to build an electronic textbook for parallel programming that exploits Caltech Archetypes model of program construction. The tools provide a way for Fortran, HPF and C++ programmers to add special annotations to source code that allow the source code to be converted automatically into WWW hypertext documents. In addition, special compiler based tools examine the source code and discover all subroutine call sites and automatically build the hypertext links to the appropriate subroutine definitions. Thus, Villacis is a system for generating textbook templates, in a hypertext document format, for source code.

The important thing to note is that Villacis operates on source code and is implemented in tools for a compiler. Source code is not executable, as is readily understood by those of ordinary skill in the art. To the contrary, a compiler and linker are necessary to convert source code into executable code. Thus, Villacis could not possibly teach the features of the present invention, i.e. "during execution of said application program, selecting an element defined in the document type definition file based on a routine called by said application program" since Villacis cannot operate on an executing application program but instead, operates on source code.

The Office Action alleges that these features of claim 6 are taught by Villacis on pages 4, 7-8 and 14. Page 4 of Villacis explicitly states that "In this section we examine the Anno utility for building web documents from source code." Again, while Villacis talks about converting source code to HTML documents, Villacis operates on source code and does not having anything to do with selecting elements defined in a document

type definition file <u>based on a routine called by an application during execution of the application</u>, as recited in claim 6.

On pages 7-8, Villacis teaches that every Fortran, C and C++ program consists of a sequence of data definitions, type definitions and function definitions. Villacis further teaches that the Mccp component of the Villacis mechanism accepts programs written in a variety of Fortran dialects and C++ flavors, and generates hypertext versions of the source code. Each program subroutine and class definition is isolated into separate files. A complete subroutine call graph is used to generation hypertext links between each invocation site and the appropriate routine definitions. In the case of C++, a class browser is built that is tailored to class definitions in the source code. The result is a forms-based HTML document tree.

Again, this section of Villacis merely serves to bolster Applicants' position that Villacis operates on <u>source code</u> and thus, cannot possible teach or even suggest the feature of selecting an element defined in a document type definition file <u>based on a routine called by an application program during execution of the application program</u>, as recited in claim 6. All Villacis teaches is the ability to generate hypertext document versions of source code. Villacis does not operate on executing applications and does not operate in an execution environment. Therefore, Villacis cannot teach or suggest selecting anything based on a routine called by an application during execution of the application, let alone selecting an element defined in a document type definition file.

Page 14 of Villacis merely further emphasizes this basic distinction of the present invention over the mechanism of Villacis. Page 14 describes the "runtime phase" (as opposed to the "compiler phase" discussed above) of the Villacis mechanism, i.e. the Mccp browser program that permits the user to browse the HTML documents produced from the source code (during the "compiler phase"). While this section of Villacis discusses the creation of new documents when a user clicks on a form button or expand link in the initial document, these operations are performed on the HTML document version of the source code. They do not select an element defined in a document type definition file based on a routine called by an application during execution of the application. To the contrary, nowhere in Villacis is either the source code or the HTML document version of the source code ever executed. This is because Villacis is only

concerned with generating templates of the source code for Villacis' "textbook" so that they may be browsed using the Mccp browser. Villacis is not concerned with executing code and furthermore, is not concerned with selecting an element from a document type definition file based on a routine called by an application during execution of the application. Thus, while Villacis may teach a runtime environment, this runtime environment is only taught as providing a browser by which to view the HTML document versions of the source code and has nothing to do with the selection of elements defined in a document type definition file based on a routine called by an application during execution of the application.

Therefore, even if Villacis were somehow combinable with Meltzer, *arguendo*, and assuming that one of ordinary skill in the art were somehow motivated to even attempt the combination, the result still would not be the invention as recited in independent claims 6, 17 and 25 because neither reference teaches or suggests selecting an element defined in a document type definition file based on a routine called by an application during execution of the application. Since neither reference teaches or suggests such features, any alleged combination of the references still would not result in these features being taught or suggested.

Moreover, it is not at all clear how the system of Villacis could be combined with the system of Meltzer. Meltzer is directed to a method of exchanging self-defining electronic documents, such as XML based documents, between trading partners without custom integration. Villacis is directed to generating HTML versions of source code that may be browsed. There is no commonality between these references other than using markup language based documents. The whole purpose of Meltzer is to provide an electronic commerce system that replaces closed trading partner networks based on proprietary standards with open markets. The purpose of Villacis is to generate a browseable "textbook" of source code. It is not clear how a commerce system for open markets can be combined with a system for generating a browseable "textbook" of source code.

Furthermore, there is no suggestion in either reference to even attempt the combination. That is, there is no deficiency noted in Meltzer for which Villacis is a solution or vice versa. This is rooted in the fact that these references are directed to

completely different fields of endeavor and would not even be considered combinable by anyone of ordinary skill in the art. To the contrary, the only basis for even alleging such a combination of references is based on an attempt to recreate the claimed invention using hindsight. While Applicants recognize that all examination includes some measure of hindsight, when the rejection is completely based on a hindsight reconstruction, it is improper. That is, while the Examiner may use the claims as a basis for locating prior art, once this prior art is found, the Examiner must take an objective view and determine whether one of ordinary skill in the art, being presented only with the references and without having a prior knowledge of Applicants' claimed invention, would necessarily combine the references in the particular manner necessary and modify them in the particular way necessary to arrive at the claimed invention. In the present case, the answer is clearly "no."

In view of the above, Applicants respectfully submit that neither Meltzer nor Villacis, either alone or in combination, teach or suggest all of the features of independent claims 6, 17 and 25. At least by virtue of their dependency on claims 6 and 17 respectively, neither Meltzer nor Villacis, either alone or in combination, teach or suggest the features set forth in dependent claims 7-11 and 18-22. Accordingly, Applicants respectfully request withdrawal of the rejection of claims 6-11, 17-22 and 25 under U.S.C. 103(a).

In addition, Meltzer and Villacis do not teach or suggest any of the specific features set forth in the dependent claims 7-11 and 18-22. The rejections set forth in the Office Action with regard to claims 7-11 and 18-22 are exactly the same as previously set forth in the Final Office Action dated September 23, 2003. All of the rejections of these claims allege that the additional features of the dependent claims are taught by Meltzer and thus, Villacis is not relied upon as teaching any of the additional features of the dependent claims. These rejections were addressed in the Appeal Brief filed February 4, 2004 and thus, are considered to be overcome for the same reasons set forth in the Appeal Brief. Therefore, Applicants' remarks in Applicants' Appeal Brief with regard to these dependent claim features and their distinctions over Meltzer are considered to still be applicable and still be persuasive. Thus, for the reasons set forth in Applicants' Appeal Brief, in addition to being dependent on independent claim 6, the dependent claims 7-11 and 18-22 are also

allowable by virtue of their specific recited features. Accordingly, Applicants
respectfully request withdrawal of the rejections of claims 7-11 and 18-22 under 35
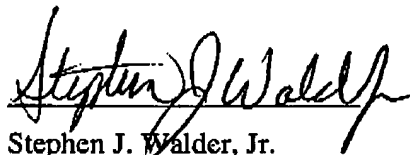U.S.C. § 103(a).

III.   **Conclusion**

It is respectfully urged that the subject application is patentable over Meltzer and
Villacis and is now in condition for allowance. The Examiner is invited to call the
undersigned at the below-listed telephone number if in the opinion of the Examiner such
a telephone conference would expedite or aid the prosecution and examination of this
application.

Respectfully submitted,

DATE: August 6, 2004

Stephen J. Walder, Jr.
Reg. No. 41,534
Yee & Associates, P.C.
P.O. Box 802333
Dallas, TX 75380
(972) 367-2001
Attorney for Applicants

Page 11 of 11
Cooper et al. – 09/306,189